Основы информационной безопасности

Лабораторная работа № 5. Исследование влияния дополнительных атрибутов

Сунгурова Мариян Мухсиновна

Содержание

# 1 Цель работы

Целью данной лабораторной работы является изучение механизмов изменения идентификаторов, применения SetUID- и Sticky-битов, а также получение практических навыков работы в консоли с дополнительными атрибутами и рассмотрение работы механизма смены идентификатора процессов пользователей, и влияние бита Sticky на запись и удаление файлов.

# 2 Теоретические сведения

При работе с командой chmod важно понимать основные права доступа, которые назначают файлам или каталогам. В Linux используется три основных типа прав доступа[1]:

* Чтение (Read) — обозначается буквой «r». Предоставляет возможность просматривать содержимое файла или каталога.
* Запись (Write) — обозначается буквой «w». Позволяет создавать, изменять и удалять файлы внутри каталога, а также изменять содержимое файла.
* Выполнение (Execute) — обозначается буквой «x». Дает разрешение на выполнение файла или на вход в каталог.

Каждый из указанных выше типов прав доступа может быть назначен трем группам пользователей:

* Владелец (Owner) — пользователь, который является владельцем файла или каталога.
* Группа (Group) — группа пользователей, к которой принадлежит файл или каталог.
* Остальные пользователи (Others) — все остальные пользователи системы.

Комбинация этих базовых прав доступа для каждой из групп пользователей определяет полный набор прав доступа для файла или каталога.

# 3 Выполнение лабораторной работы

Проверим установлен ли компилятор gcc, а также отключим SELinux(рис. fig. 1)

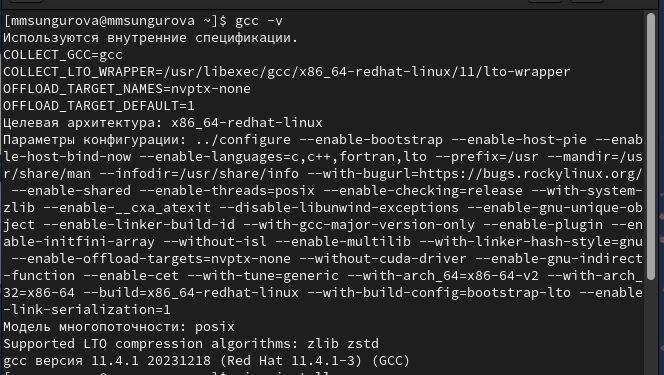


Рис. 1: Подготовка лабораторного стенда

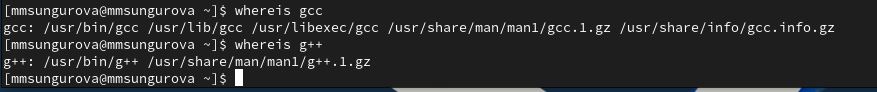


Рис. 2: Подготовка лабораторного стенда

Войдем в систему от имени пользователя guest и создадим программу simpleid.c, которая выводит идентификатор пользователя и группы(рис. fig. 3)



Рис. 3: Текст программы simpleid.c

Теперь скомпириуем программу с помощью gcc, затем, запустив её, увидим, что она выводит идентификаторы пользователя и группы 1001 и 1001 для guest, что совпадает с выводом команды id(рис. fig. 4)

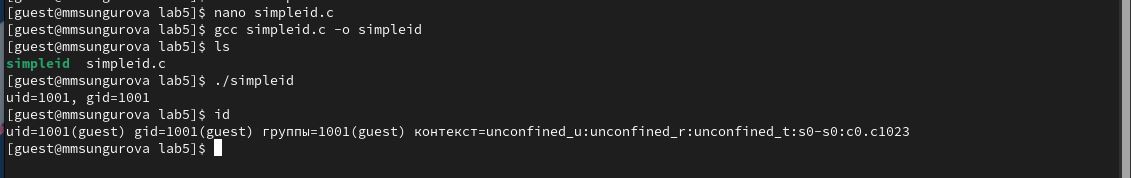


Рис. 4: Запуск программы simpleid

Усложним программу, добавив вывод действительных идентификаторов(рис. fig. 5).

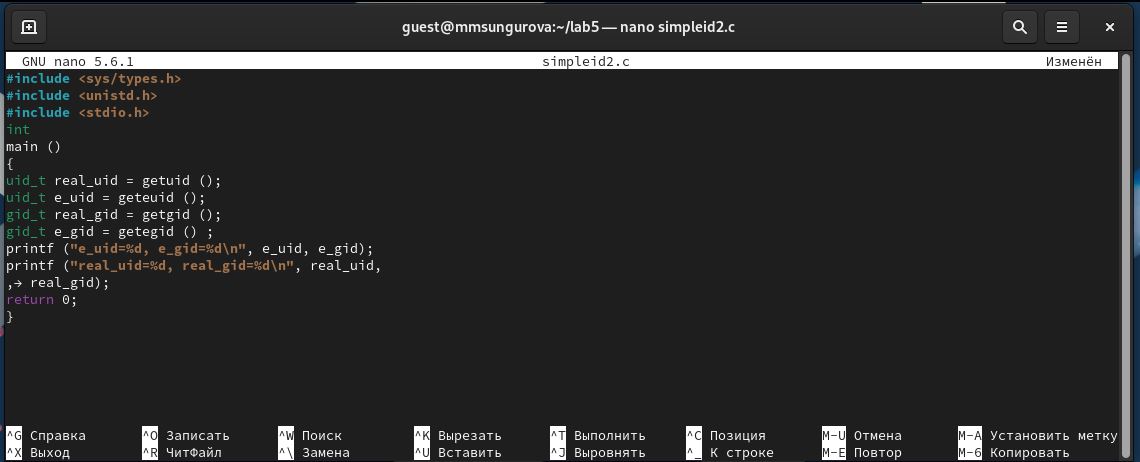


Рис. 5: Текст программы simpleid2.c

Теперь скомпириуем программу с помощью gcc, затем, запустив её, увидим, что она выводит идентификаторы пользователя и группы 1001 и 1001 для guest, что совпадает с выводом команды id(рис. fig. 6).

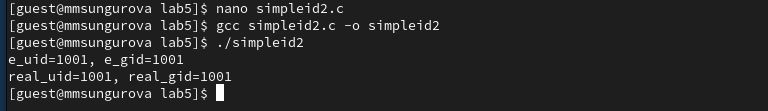


Рис. 6: Запуск программы simpleid2

От имени суперпользователя изменим владельца файла /home/guest/simpleid2 и установим SetUID-бит. Проверим корректность установленных прав и опять запустим simpleid2(рис. fig. 7).

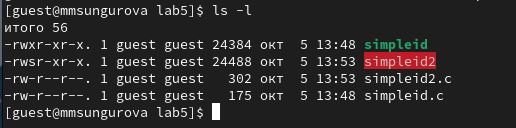


Рис. 7: Изменение владельца и запуск программы simpleid2 с установленным SetUID-битом

Проделаем аналогичные действия относительно SetGID-бита(рис. fig. 8):

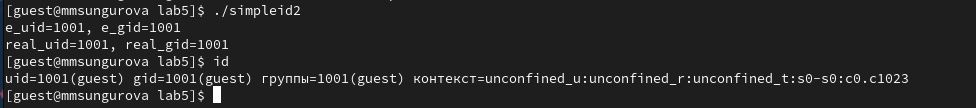


Рис. 8: Запуск программы simpleid2 с установленным SetGID-битом

Создадим программу для чтения файлов readfile.c(рис. fig. 9):

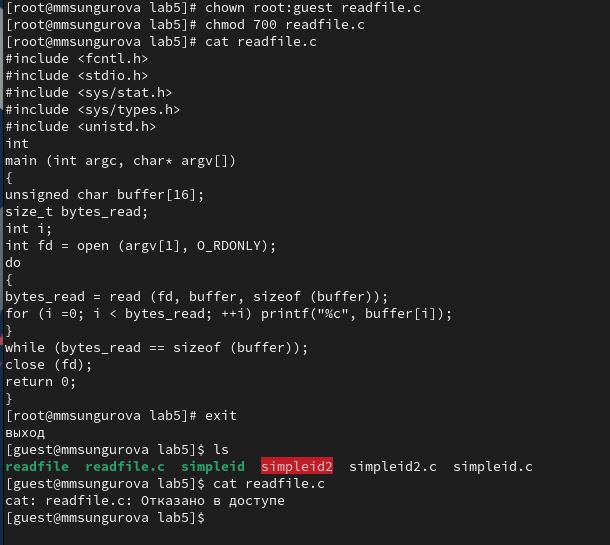


Рис. 9: Текст программы readfile.c

Скомпилируем её и сменим владельца у файла с текстом программы, затем изменим права так, чтобы только суперпользователь (root) мог прочитать его, и проверим корректность настроек(рис. fig. 9):

Сменим у программы readfile владельца и установим SetUID-бит. Теперь эта программа может прочитать файл readfile.c даже с пользователя guest, также она может прочитать файл /etc/shadow, владельцем которого guest также не является, так как программа readfile теперь имеет все права пользователя root(рис. fig. 10):

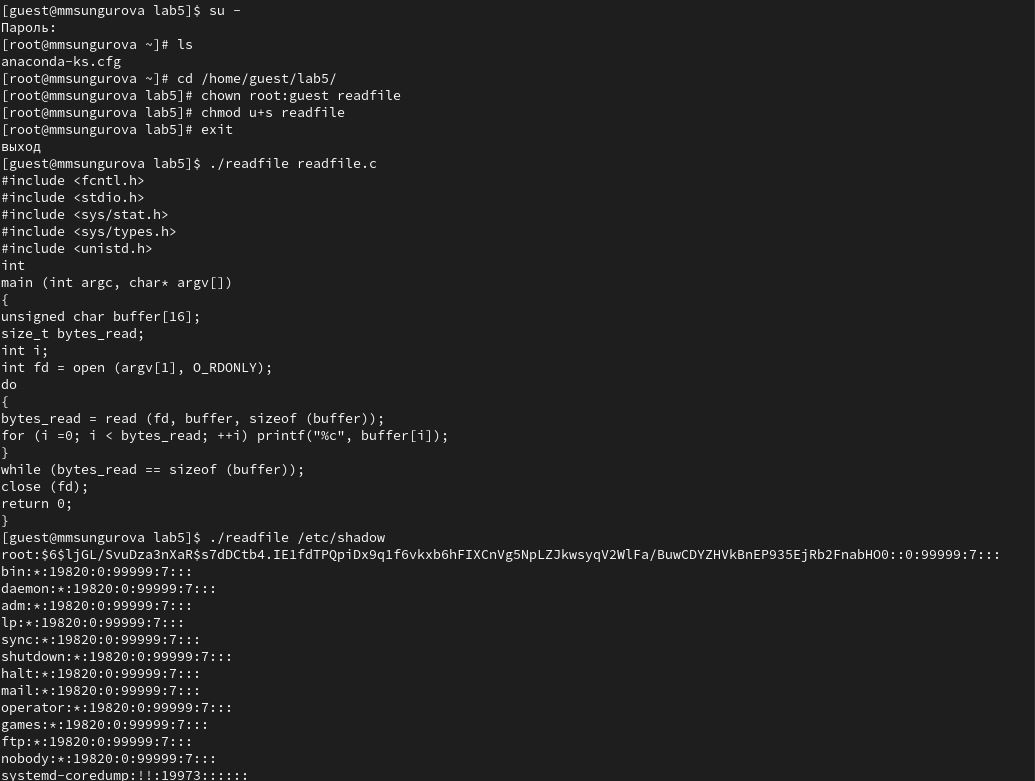


Рис. 10: Установка SetUID-бита на исполняемый файл readfile и проверка прав

После завершения установки операционной системы корректно перезапустим виртуальную машину и при запросе примем условия лицензии.

Проверим, что установлен атрибут Sticky на директории /tmp(в конце стоит t). Затем от имени пользователя guest создадим файл file01.txt в директории /tmp со словом test, затем просмотрим атрибуты у только что созданного файла и разрешим чтение и запись для категории пользователей «все остальные». После этого от пользователя guest2 попробуем дозаписать в этот файл новое слово, однако получим отказ, также нам отказано в перезаписи и удалении этого файла. Если же убрать Sticky бит, то нам будет разрешено удаление этого файла(рис. fig. 11):

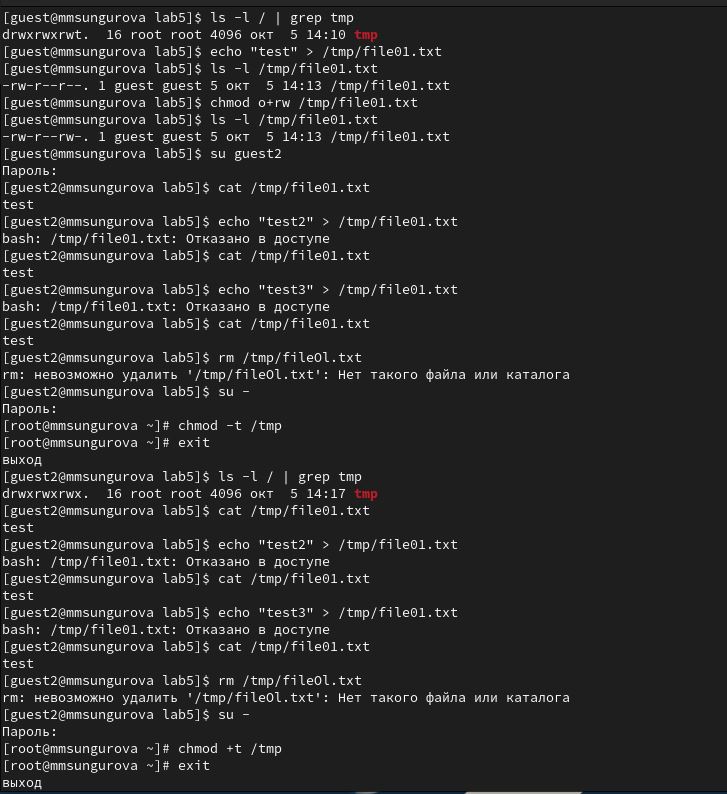


Рис. 11: Подключение образа диска дополнений

# 4 Выводы

В результате выполнения данной лабораторной работы работы были рассмотрены:

* Механизмы изменения идентификаторов, применения SetUID- и Sticky-битов.
* Получение практических навыков работы в консоли с дополнительными атрибутами.
* Механизм смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

# Список литературы

1. Граннеман С. Скотт Граннеман: Linux. Карманный справочник. 2-е изд. Вильямс, 2019. 464 с.